REMARKS/ARGUMENTS

Prior to entry of this amendment, claims 1-21 were pending in the application. An Office Action mailed February 21, 2007, rejected claim 19 under 35 U.S.C. § 112, first paragraph, as being of undue breadth. The Office Action rejected claims 1-21 under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. The Office Action rejected claim 20 under 35 U.S.C. § 101 because the claimed invention is allegedly directed to non-statutory subject matter. And finally the Office Action rejected claims 1-21 under 35 U.S.C. § 103(a) as being unpatentable over US 2003/0046307 to Rivette et al. (hereinafter "Rivette") in view of U. S. Pat. No. 6,049,811 to Petruzzi et al. (hereinafter "Petruzzi"). This amendment amends claims 1, 19, 20 and 21, but does not add or cancel any claims. Hence, after entry of this amendment, claims 1-21 will remain pending for examination. Claims 1, 19 and 20 are independent claims.

Claim Amendments

Claim 1 has been amended to recite, "the association between the first set of references and the first case file indicates that the first set of references . . . should be disclosed to a patent office in connection with the first patent application," and to recite, "the association between the first set of references and the first case file indicates that the first set of references . . . should be disclosed to a patent office in connection with the first patent application." Claim 1 has also been amended to recite that the "second interface element is configured to allow the user to . . . indicate that the first reference should be disclosed to a patent office in connection with the first patent application." Claims 19 and 20 have been amended in similar fashion. Support for these amendments can be found throughout the patent application, including, inter alia, at p. 8, ll. 24-25 and p. 31, ll. 22-25.

Claim 1 has also been amended, for clarity, to recite "displaying, at the computer, a list of the second set of references" and to more clearly recite that "displaying the list of the

second set of references comprises displaying an identifier corresponding to a first reference within the second set of references." Claims 19 and 20 have been amended in similar fashion.

Claim 21 has been amended to recite, "creating the electronic information disclosure statement comprises extracting the information from the first reference via an automated process and storing the information in the plurality of fields in the electronic information disclosure statement via the automated process." Support for this amendment can be found throughout the application, including, inter alia, at p. 6, ll. 8-27, p. 9, ll. 4-15, p. 11, ll. 13-18, p. 14, l. 9 – p 17, l. 8, and p 25, ll. 16-17.

Claims 19 and 20 have also been amended to address informalities. Specifically, the element label (k) has been replaced with the element label (h) for consistency with the remainder of the claims.

Rejections under § 112

1. Rejection of claim 19 under § 112, ¶ 1

The office action rejected claim 19 under the first paragraph of § 112 as being of undue breadth Specifically, the office action asserted that claim 19 is a "single means" claim. This rejection is respectfully traversed, and it is respectfully submitted that the office action's finding is unsupported, for at least two reasons.

First, the case on which the rejection relies, *In re Hyatt*, 708 F.2d 712, 218 USPQ 195 (Fed. Cir. 1983) defines a "single means claim" to be "a claim drafted in 'means-plus-function' format yet reciting only a single element instead of a combination." 708 F.2d at 713. The MPEP provides a substantially similar definition: "A single means claim, i.e., where a means recitation does not appear in combination with another recited element of means . . . ." MPEP § 2164.08(a). In contrast, claim 19 does not recite any "means-plus-function" elements. Instead, it recites a processor and a computer readable memory, in addition to several instructions encoded on the computer readable medium. Accordingly, there is no basis for describing claim 19 as a single means claim.

Second, even assuming one of the elements recited by claim 19 might be considered a "means-plus-function" element (and none of the elements reasonably can be construed this way), claim 19 recites ten different elements in combination (the processor, the computer readable medium, and eight instruction elements). For at least this additional reason, claim 19 properly cannot be rejected as a single means claim. Reconsideration of the rejection of claim 19 is respectfully requested.

2. Rejections of claim 1-21 under § 112, ¶ 2

The office action also rejected claims 1-21 under the second paragraph of § 112, on a variety of grounds. First, the office action objected to the term "a first interface element," as recited in claims 1, 19 and 20. The office action also objected to the term "wherein the display comprises an identifier," as recited in those claims. Finally, the office action objected to the term "marker," as used in claim 4.

a) "First interface element"

Claims 1, 19 and 20 each recite "providing [or instructions to provide] a first interface element that is configured to allow a user to invoke a display of a list of the second set of references." The office action evidences a belief that the term "first interface element" is unclear, asking "What is an 'interface element'? Is it a GUI? If so, how can the input correspond to the first interface element?" Office Action, at 3. The applicant respectfully disagrees that the term "first interface element" lacks clarity and therefore traverses the rejections on this basis.

Regarding the standard for compliance with § 112, ¶ 2, the MPEP instructs that "[t]he focus during examination of claims for compliance with the requirement for definiteness of 35 U.S.C. 112, second paragraph, is whether the claim meets the threshold requirements of clarity and precision, not whether more suitable language or modes of expression are available." MPEP § 2173.02. Further, " Definiteness of claim language must be analyzed, not in a vacuum, but in light of: (A) The content of the particular application disclosure; (B) The teachings of the prior art; and (C) The claim interpretation that would be given by one possessing the ordinary level of skill in the pertinent art at the time the invention was made." *Id.*

The applicants submit that the term, "interface element" is known in the art as an element (also referred to as a "widget") of a graphical user interface that provides some interaction between the user and the graphical user interface, either by displaying information for the user, or by receiving information from the user. For example, wikipedia notes, "A graphical user interface (GUI) is a type of user interface which allows people to interact with a computer and computer-controlled devices which employ graphical icons, visual indicators or special graphical elements called 'widgets', along with text, labels or text navigation to represent the information and actions available to a user. The actions are usually performed through direct manipulation of the graphical elements." Wikipedia, "Graphical User Interface," <http://en.wikipedia.org/wiki/Graphical_user_interface> (downloaded (copy attached hereto as Exhibit A).

In this case, the applicant's disclosure clearly discloses an examples of interface elements that allow a user to invoke a display of a list of references. See, for example, Figs. 16 and 17, along with lines 6-34 on page 32 of the Application. This disclosure teaches that

> A user may select one or more of the cases for cross referencing by mouse clicking select boxes 1614 and 'Add to Selected List' button 1612. . . .
>
> Fig. 17 illustrates an add group reference Web page 1700 useful for associating electronic documents from one case to another case according to one embodiment of the present invention. Add group reference page 1700 displays the electronic reference documents associated with the cases in the selected case list of Fig. 16. (Application, p. 32, ll. 12-14, 22-25.)

Hence, in the example disclosed by the application, the graphical user interface provides the user with one or more checkboxes and a button (all of which are "interface elements") to invoke a display of a list of a set of references. The applicants respectfully submit that one skilled in the art, upon reading the disclosure of the application, would clearly understand what is meant by the term "interface element," as that term is used in claims 1, 19 and 20. Reconsideration of the rejections of claims 1-21 under § 112, ¶ 2, therefore, is requested.

b) "Wherein the display comprises an identifier "

The office action also objected to the term "displaying [or instructions to display] a list of the second set of references, wherein the display comprises an identifier corresponding to a first reference within the second set of references." Claim 1 has been amended to recite

"displaying, for the user, a list of the second set of references, wherein displaying the list of the second set of references comprises displaying an identifier corresponding to a first reference within the second set of references." Claims 19 and 20 have been amended in similar fashion, and it is believed that these amendments overcome the rejections on these grounds. Reconsideration of the amended claims is respectfully requested.

c) "Marker"

The office action also objected to the term "marker," as that term is used in claim 4. As illustrated by Fig. 18, and described at lines 12-23 on page 33 of the application, the marker is a visual indicator on a user interface that indicates an association between a particular reference (e.g., prior art document) and case (e.g., patent application). Figure 18 of the application illustrates a user interface (in this case, a web page 180) with a set of columns and rows, as recited by claim 4. Each row in Fig. 18 includes a "marker" (in this case, an "X") to indicate that a reference corresponding to a particular row is associated with a case corresponding to a particular column. *See also* Application, p. 18, ll. 12-23. While other embodiments might feature other types of markers, the disclosed examples clearly teach what is meant by the term "marker," as it is used in claim 4. Hence, it is submitted that one skilled in the art, upon reading the disclosure of the application, would understand what is meant by the term "marker," and that claim 4 therefore complies with § 112, ¶ 2. Reconsideration of the rejection of claim 4 is respectfully requested.

Rejection under § 101

The office action rejected claim 20 under § 101. This rejection is also respectfully traversed. As grounds for the rejection, the office action asserts, "[c]laim 20, as written, is claiming only instructions. Thus, the claim is not statutory." Office Action, at 6. The applicant respectfully disagrees. Rather, claim 20 recites, "[a] computer program embodied on a computer readable medium, the computer program comprising a set of instructions executable by one or more computers." According to the MPEP, "a claimed computer-readable medium encoded with a computer program is a computer element which defines structural and functional

interrelationships between the computer program and the rest of the computer which permit the computer program's functionality to be realized, and is thus statutory." MPEP § 2106.01. Claim 20 recites a computer program embodied on a computer readable medium, and is therefore directed toward statutory subject matter. Reconsideration of claim 20 is respectfully requested.

Rejections under § 103

The office action ejected claims 1-21 under 35 U.S.C. § 103(a) as being unpatentable over Rivette, in view of Petruzzi. These rejections are respectfully traversed.

Rivette generally discloses a system for maintaining databases of patents (and non-patent information) and for processing groups of patents (Rivette, abs.) Rivette, however, contains little (if any disclosure) relating to the prosecution of pending applications – instead it is focused on dealing with issued patents. Accordingly, Rivette has no need to deal with information disclosure statements, which, one skilled in the art would appreciate, generally are pertinent only to pending applications, not issued patents. (Indeed, a text search of the Rivette references fails to identify a single mention of an "information disclosure statement").

In contrast, the pending claims are directed to tools for associating prior art references with pending applications, so that the references can be disclosed to a patent office. It is perhaps unsurprising, then, that Rivette fails to teach or suggest several elements of even the independent claims in the application.

Consider, for example, claim 1. Claim 1 has been amended to recite that "the association between the first set of references and the first case file indicates that each of the references in the first set of references are relevant to the first patent application, and should be disclosed to a patent office in connection with the first patent application." Rivette teaches nothing about disclosing any documents to a patent office, let alone maintaining an association between a set of references and a case file (which corresponds to a patent application) to indicate that the references should be disclosed to a patent office in connection with the patent application.

Similarly, claim 1, as amended, recites, "providing a second interface element that is configured to allow the user to . . . indicate that the first reference should be disclosed to a

patent office in connection with the first patent application." Rivette also fails to disclose this element. Moreover, since Rivette does not even contemplate disclosing references to a patent office, there would be no need, in Rivette's system, for any type of interface element to allow a user to indicate that a reference should be disclosed.

In fact, the office action does not even identify any disclosure in Rivette that might be interpreted to teach or suggest "providing [an] interface element that is configured to allow the user to identify [a] first [prior art] reference as being relevant to [a] first patent application," as recited by the previous version of claim 1.

Petruzzi, for its part, briefly discloses creating an information disclosure statement. Petruzzi, c. 4, l. 48 -- c. 6, l. 7. However, Petruzzi does not disclose any of the other elements missing from Rivette, as described above. Moreover, because Petruzzi teaches that the information disclosure statement is created only by user input and has nothing to do with coordinating reference information between difference cases, there is no way to read Petruzzi as suggesting any of the other procedures recited by claim 1. Accordingly, the combination of Petruzzi and Rivette collectively fails to teach or suggest each element of claim 1, and claim 1 therefore is believed to be allowable over that combination.

Independent claims 19 and 20 each recite elements similar to those of claim 1, and they are believed to be allowable for at least similar reasons. Dependent claims 2-18 and 21 each ultimately depend from claim 1 and therefore are believed to be allowable at least by virtue of their dependence from that claim.

Moreover, many of the dependent claims recite additional novel features and are independently allowable based on those recitations. Merely by way of example, claim 21 recites, inter alia, "creating the electronic information disclosure statement comprises extracting the information from the first reference via an automated process and storing the information in the plurality of fields in the electronic information disclosure statement via the automated process." The office action fails even to address claim 21, and it is believed that nothing in either Rivette or Petruzzi teaches these additional elements.

CONCLUSION

In view of the foregoing, the Applicants believe all claims now pending in this application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

If the Examiner continues to believe that the claims are unpatentable over Rivette (taken either alone or in combination with any other reference) or continues to have any trouble understanding the meaning of the claims, the undersigned respectfully requests a telephone interview to discuss the rejections. The undersigned can be reached by telephone at 303-571-4000 to schedule the telephone interview.

Respectfully submitted,

Date: August 21, 2007

/Chad E. King/
Chad E. King
Reg. No. 44,187

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, CA 94111-3834
Tel: 303-571-4000 (Denver office)
Fax: 303-571-4321 (Denver office)

CEK/jep

61130622 v1

EXHIBIT A

Graphical user interface

From Wikipedia, the free encyclopedia

A **graphical user interface (GUI)** is a type of user interface which allows people to interact with a computer and computer-controlled devices which employ graphical icons, visual indicators or special graphical elements called "widgets", along with text, labels or text navigation to represent the information and actions available to a user. The actions are usually performed through direct manipulation of the graphical elements. Use of this acronym led to creation of the neologism *guituitive* (an interface which is intuitive).

Contents

History

Precursors to graphical user interfaces

The precursor to graphical user interfaces was invented by researchers at the Stanford Research Institute, led by Douglas Engelbart. They developed the use of text-based hyperlinks manipulated with a mouse for the On-Line System. The concept of hyperlinks was further refined and extended to graphics by researchers at Xerox PARC, who went beyond text-based hyperlinks and used a GUI as the primary interface for the Xerox Alto computer. Most modern general-purpose GUI's are derived from this system. As a result, some people call this class of interface a PARC User Interface (PUI) (note that PUI is also an acronym for perceptual user interface).

PARC User Interface

The PARC User Interface consists of graphical widgets (often provided by widget toolkit libraries) such as windows, menus, radio buttons, check boxes and icons. The PARC User Interface employs a pointing device in addition to a keyboard. These aspects can be emphasized by using the alternative acronym WIMP, which stands for Windows, Icons, Menus and Pointing device.

Evolution of graphic user interfaces

Following PARC the first commercially successful GUI-centric computer operating models were those of the Apple Lisa but more successfully that of Macintosh System graphical environment. The graphical user interfaces familiar to most people today are Microsoft Windows, Mac OS X, and the X Window System interfaces. IBM and Microsoft used many of Apple's ideas to develop the Common User Access specifications that formed the basis of the user interface found in Microsoft Windows, IBM OS/2 Presentation Manager, and the Unix Motif toolkit and window manager. These ideas evolved to create the interface found in current versions of the Windows operating system, as well as in Mac OS X and various desktop environments for Unix-like systems. Thus most current graphical user interfaces have largely common idioms.

Graphical user interface design

Graphical user interface design is an important adjunct to application programming. Its goal is to enhance the usability of the underlying logical design of a stored program. The visible graphical interface features of an application are sometimes referred to as "chrome". They include graphical elements (widgets) that may be used to interact with the program. Common widgets are: windows, buttons, menus, and scroll bars. Larger widgets, such as windows, usually provide a frame or container for the main presentation content such as a web page, email message or drawing. Smaller ones usually act as a user-input tool.

An example of KDE, one of the X Window System's many graphical user interfaces available for Unix-like systems

The widgets of a well-designed system are functionally independent from and indirectly linked to program functionality, so the graphical user interface can be easily customized, allowing the user to select or design a different *skin* at will. See Model-view-controller for more information.

See also: GUI Design Principles

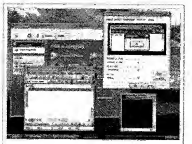
Screenshot of Mac OS X's Graphical User Interface. Note the usage of a single menu bar in different programs.

Zooming user interface

Many research groups in North America and Europe are currently working on the Zooming User Interface (ZUI) which is a logical advancement on the graphical user interface, blending some three-dimensional movement with two-dimensional or "2.5D" vector objects.

Some graphical user interfaces are designed for the rigorous requirements of vertical markets. These are known as "application specific graphical user interfaces." Examples of application specific graphical user interfaces:

- Touchscreen point of sale software used by waitstaff in busy restaurants
- Self-service checkouts used in some retail stores..
- ATMs



Screenshot of Windows XP's Graphical User Interface with four windows open.

- Airline self-ticketing and check-in
- Information kiosks in public spaces like train stations and museums
- Monitor/control screens in embedded industrial applications which en system (RTOS).

The latest cell phones and handheld game systems also employ application s graphical user interfaces.

Screenshot of the GNOME [illegible] applications Totem, Gedit, Gnome Terminal, Epiphany, GAIM and Nautilus are all applications specifically [illegible] desktop environment.

Graphical user interfaces compared to command line interfaces

Graphical user interfaces were introduced in reaction to the steep learning curve of command-line interfaces (CLI), which require commands to be typed on the keyboard. Since the commands available in command line interfaces can be numerous, complicated operations can be completed using a short sequence of words and symbols. This allows for greater efficiency and productivity once many commands are learned, but reaching this level takes some time because the command words are not easily discoverable. WIMPs ("window, icon, menu, pointing device"), on the other hand, present the user with numerous widgets that represent and can trigger some of the system's available commands.

WIMPs extensively use modes as the meaning of all keys and clicks on specific positions on the screen are redefined all the time. CLIs use modes only in the form of a current directory.

Most modern operating systems provide both a graphical user interface and some level of a CLI, although the graphical user interfaces usually receive more attention. The graphical user interface is usually WIMP-based, although occasionally other metaphors surface, such as those used in Microsoft Bob, 3dwm or File System Visualizer (FSV).

Applications may also provide both interfaces, and when they do the graphical user interface is usually a WIMP wrapper around the command-line version. This is especially common with applications designed for Unix-like operating systems. The latter used to be implemented first because it allowed the developers to focus exclusively on their product's functionality without bothering about interface details such as designing icons and placing buttons. Designing programs this way also allows users to run the program non-interactively, such as in a shell script.

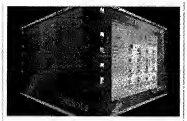
Graphical user interfaces compared to text user interfaces

Three-dimensional user interfaces

For typical computer displays, *three-dimensional* is a misnomer—their displays are two-dimensional. Three-dimensional images are projected on them in two dimensions. Since this technique has been in use for many years, the recent use of the term three-dimensional must be considered a declaration by equipment marketers that the speed of three dimension to two dimension projection is adequate to use in standard graphical user interfaces.

Three-dimensional graphical user interfaces are common in science fiction literature and movies, such as in *Jurassic Park*, which features Silicon Graphics' three-dimensional file manager, "File system navigator", an actual file manager that never got much widespread use as the user interface for a Unix computer.

In science fiction, three-dimensional user interfaces are often immersible environments like William Gibson's Cyberspace or Neal Stephenson's Metaverse. Three-dimensional graphics are currently



Screenshot showing the 'cube' plugin of Compiz on Ubuntu

mostly used in computer games, art and computer-aided design (CAD). There have been several attempts at making three-dimensional desktop environments like Sun's Project Looking Glass or SphereXP (http://www.spheresite.com/) from Sphere Inc. A three-dimensional computing environment could possibly be used for collaborative work. For example, scientists could study three-dimensional models of molecules in a virtual reality environment, or engineers could work on assembling a three-dimensional model of an airplane. This is a goal of the Croquet project and Project Looking Glass by Java. [1]

The use of three-dimensional graphics has become increasingly common in mainstream operating systems, but mainly been confined to creating attractive interfaces—eye candy—rather than for functional purposes only possible using three dimensions. For example, user switching is represented by rotating a cube whose faces are each user's workspace, and window management is represented in the form of Exposé on Mac OS X, or via a Rolodex-style flipping mechanism in Windows Vista. In both cases, the operating system transforms windows on-the-fly while continuing to update the content of those windows.

Interfaces for the X Window System have also implemented advanced three-dimensional user interfaces through compositing window managers such as Beryl and Compiz using the AIGLX or XGL architectures, allowing for the usage of OpenGL to animate the user's interactions with the desktop.

Another branch in the three-dimensional desktop environment is the three-dimensional graphical user interfaces that take the desktop metaphor a step further, like the BumpTop, where a user can manipulate documents and windows as if they were "real world" documents, with realistic movement and physics. With the current pace on three-dimensional and related hardware evolution, projects such these may reach an operational level soon.

See also

- Ajax
- Apple v. Microsoft
- Ergonomics
- Human-Machine Interface
- Live User Interface
- Look and feel
- Model-view-controller
- Object-oriented user interface
- Post-WIMP
- Rich Internet applications

- Skin
- User interface engineering
- Vector-Based GUI
- WIMP (computing)

References

1. ^ Project Looking Glass (http://www.sun.com/software/looking_glass/)

Franco Kipkemboi

External links

- Graphical User Interface Gallery (http://toastytech.com/guis/index.html), screenshots of various Graphical User Interfaces
- Marcin Wichary's GUIdebook (http://www.guidebookgallery.org/), Graphical User Interface gallery: over 5500 screenshots of GUI, application and icon history
- The Real History of the GUI (http://www.sitepoint.com/article/real-history-gui), a very interesting article by Mike Tuck
- A History of the GUI (http://arstechnica.com/articles/paedia/gui.ars), by Jeremy Reimer of Ars Technica

Elements of graphical user interfaces

About box · Accordion · Balloon help · Button · Check box · Combo box · Combutcon · Context menu · Dialog box · Disclosure widget · Drop-down list · File dialog · Grid view · Heads-up display · Icon · Label · List box · Menu · Menu bar · Pie menu · Progress bar · Radio button · Ribbon · Scrollbar · Sidebar · Slider · Spinner · Status bar · Tab · Text box · Toolbar · Tooltip · Tree view · Widget

Categories: Wikipedia articles needing rewrite | Articles with sections needing expansion | User interface | Graphical user interface | Software architecture